



4D v1x ODBC Driver

INSTALLATION GUIDE

Open Database Connectivity Standard (ODBC) is an open standard application programming interface (API) used to access one or more databases from a single application.

In order to set up a connection via ODBC between an application and a database, you will need three things:

- An ODBC compliant application (in our case 4D).
- An ODBC compliant database (ORACLE, MS SQL or even a 4D database).
- An ODBC driver for that particular database (in our case 4D v1x ODBC Driver, where *v1x* stands for v11, v12, v13...).

An ODBC connection involves an application, a driver manager or a configuration tool (such as ODBC Administrator), a driver for the database or database management system (DBMS) and a database.

More information about ODBC can be found at the following addresses:

<http://www.iodbc.org/>

<http://msdn2.microsoft.com/en-us/library/ms712628.aspx>

Installing the 4D ODBC Driver

The 4D ODBC Driver for 4D Server exists for both Mac and PC platforms (32 bits and 64 bits). It is based on ODBC Driver Manager 3.5 and does not require any external libraries: everything is inside.

On a 64-bit operating system

If you have installed the 32-bit Installer on your 64-bit OS, you must use the 32-bit ODBC Data Source Administrator instead of the default 64-bit one. You will retrieve the appropriate administrator at the following location:

C:\Windows\SysWOW64\odbcad32.exe

If you have installed the 64-bit Installer, you access the default administrator via the Start menu (Start/Control Panel/Administrative Tools/Data Sources (ODBC)).

For more information regarding the two ODBC Data Source Administrators, please refer to the following Microsoft technical note: <http://support.microsoft.com/kb/942976/en-us>

On a Macintosh Platform

The 4D ODBC Driver on Macintosh must be installed manually. You will get a folder containing two elements:

- 4D vXX ODBC-Driver.bundle
- OpenSSL.framework folder

... where XX is the 4D ODBC Driver version: 12 for v12, 13 for v13 and so on.

The procedure to install the ODBC Driver manually is the following:

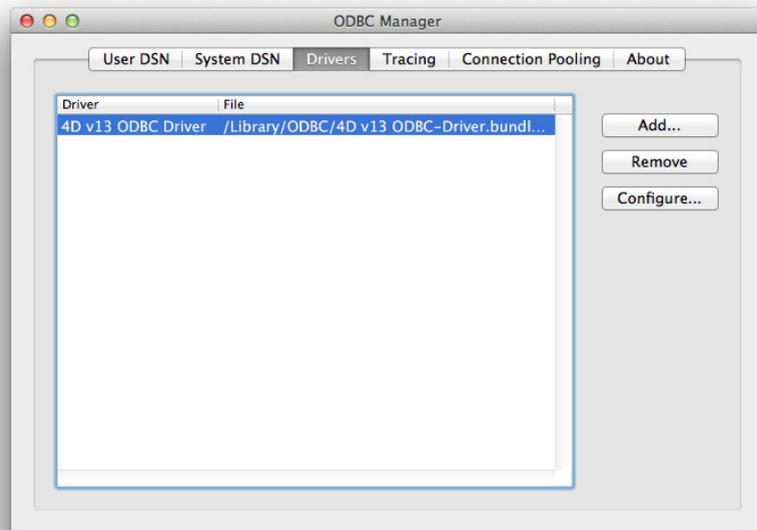
- 1 Place the OpenSSL.framework folder into the {Library}/Frameworks/ location (you must enter the Administrator password)**
- 2 Place the 4D vXX ODBC-Driver.bundle package into the {Library}/ODBC/ location (XX is the 4D ODBC Driver version: 12 for v12, 13 for v13 and so on).**
- 3 At the {Library}/ODBC/ location create a text file named *odbcinst.ini* and copy the following code into this file (replace first "XX" with the 4D ODBC driver version: 12 for v12, 13 for v13 and so on):**

```
[ODBC Drivers]
4D vXX ODBC Driver = Installed
[4D vXX ODBC Driver]
Driver             = /Library/ODBC/4D vXX ODBC-Driver.bundle/Contents/MacOS/4D vXX ODBC-Driver
Setup              = /Library/ODBC/4D vXX ODBC-Driver.bundle/Contents/MacOS/4D vXX ODBC-Driver
APILevel           = 2
ConnectFunctions   = YYN
DriverODBCVer      = 3.52
FileUsage           = 0
SQLLevel           = 3
```

You can now launch the ODBC Administrator in order to create a Data Source Name (DSN). To add a DSN, you have to launch the ODBC Administrator located in the "Applications/Utilities/" folder.

Note : On Mac OS X 10.6 Snow Leopard, the ODBC Administration Tool is no longer provided as standard. However, it can be downloaded from here:
http://support.apple.com/downloads/ODBC_Administrator_Tool_for_Mac_OS_X

By clicking the "Drivers" tab, you can check that the 4D ODBC Driver is correctly installed:



Note : Under Mac, in order to work with Microsoft Excel with any external data sources, you will need to install the OpenLink framework.

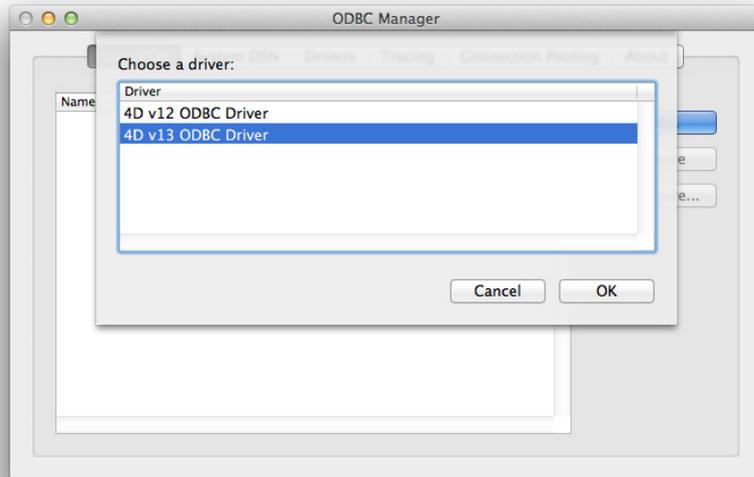
To add a DSN using the 4D ODBC Driver, you have to carry out the following steps:

1 Click on the following tab:

- "User DSN" (this makes the data source available only to the user currently logged on to this computer).

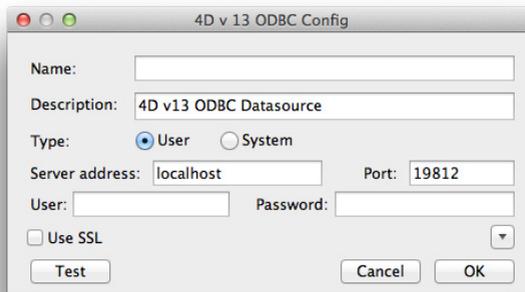
Note : On the Mac Snow Leopard OS, we recommend that you use a System DSN instead.

- 2 Click the **Add** button and choose a 4D v1x ODBC Driver from the list of configured drivers:



- 3 Click **OK**.

The 4D ODBC Driver's own configuration panel appears:



- 4 Click **OK** to save any changes before you close the configuration panel window.

Note : You may encounter the following anomalies when creating a DSN via the Mac OS X ODBC Administrator:

- When you create a new DSN, you may not see it right away and may need to quit and restart the Mac OS X Administrator in order for it to appear.
- Even if you add a new ODBC User data source, the ODBC Driver will

create it as a System DSN.

These anomalies are both due to bugs in the Mac OS X ODBC Administrator and have nothing to do with the 4D ODBC Driver or Installer.

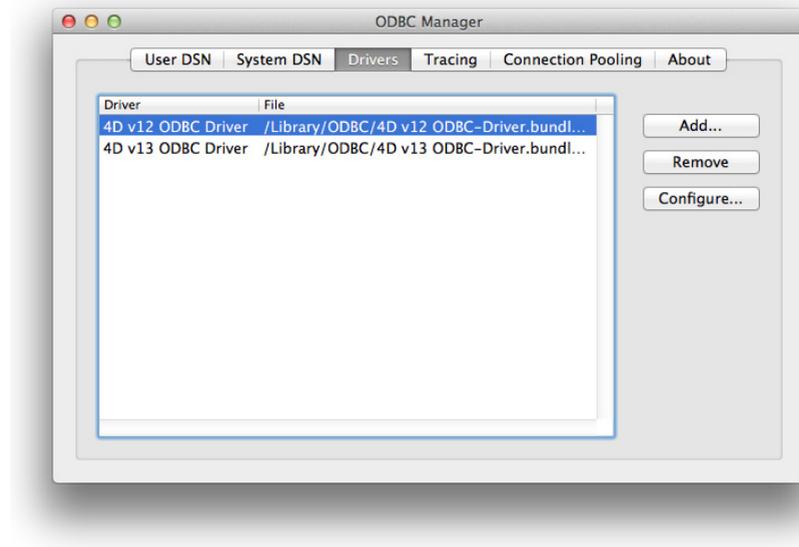
Uninstalling on Macintosh Platforms

To uninstall a 4D v1x ODBC Driver, you must use the ODBC Administrator, which can be found in the Application Utilities folder.

Note : On Mac OS 10.6 Snow Leopard, the ODBC Administrator Tool is no longer provided as standard. However, it can be downloaded from here:

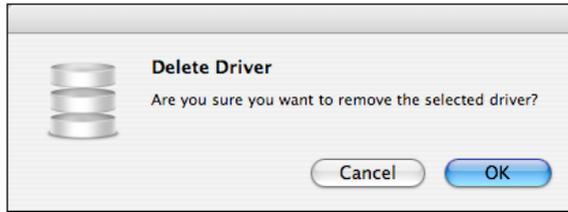
http://support.apple.com/downloads/ODBC_Administrator_Tool_for_Mac_OS_X

Launch the ODBC Administrator and click on the Drivers tab:

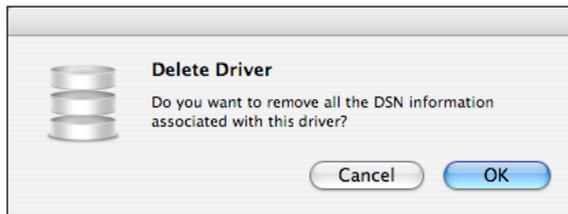


In order to make further changes, be sure that you have adequate rights by taking a look at the padlock in the bottom left of the dialog box. If it is locked, you will need to unlock it by entering the Administrator password.

To remove a 4D v1x ODBC Driver, just select it in the list of installed drivers and press the **Remove** button. You will be asked to confirm this removal:

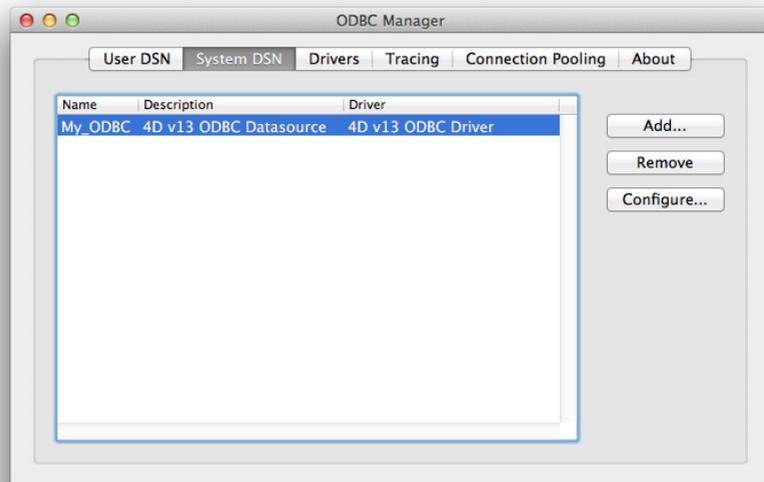


Then you will be asked if you want to also remove all the information regarding the DSN associated with the ODBC driver:



If you intend to install a new ODBC driver, you can click on **Cancel** to keep the old DSN information.

In any case, you can remove the DSN entry later by clicking on the System DSN tab, selecting the DSN in the list and clicking on **Remove**:



You will then have to remove the following files manually:

/Library/4D ODBC-Driver.bundle

/Library/Frameworks/OpenSSL.framework

On a PC Platform

In order to install the 4D v1x ODBC Driver on Windows, you simply need to run the 4D v1x SQL Installer.

4D ODBC Driver 32 bits

The installer is provided with the 4D Installer:

- If you choose the "4D Easy Install" or "4D Easy Install+Server" option, the driver is automatically installed in the C:\Program Files\4D\4Dv1x.x\ folder.
- If you choose the "Custom" installation, then you will need to check the "ODBC Driver" check box. The driver is installed in the C:\Program Files\4D\4Dv1x.x custom\ folder.

4D ODBC Driver 64 bits

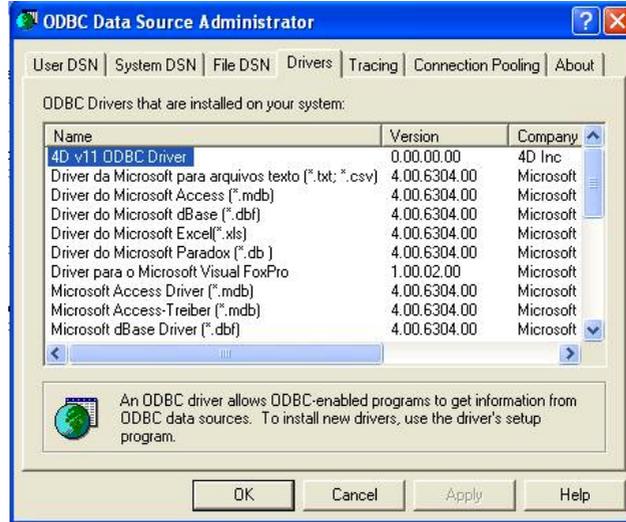
You need to download directly the 4D ODBC Driver 64 bits from the 4D Web site (<http://www.4d.com/downloads/products.html>).

If there is already a previous version of the driver installed on the target machine, the 4D v1x ODBC Driver Installer will automatically upgrade it to the newer version. If there is a newer version of the driver already installed on the machine and you want to downgrade to an older one, you must first uninstall the more recent version before installing the older one.

After running the ODBC Driver Installer, you need to define a DSN in order to connect to the desired database.

To add a DSN, you have to launch the ODBC Data Source Administrator by selecting Start/Control Panel/Administrative Tools/Data Sources (ODBC).

By clicking the "Drivers" tab, you can check that the 4D v1x ODBC Driver is correctly installed:



To add a DSN using the 4D v1x ODBC Driver, you have to carry out the following steps:

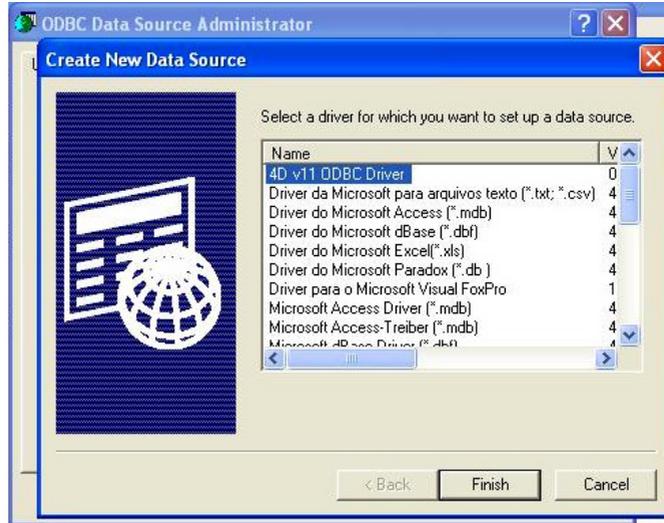
1 Click on one of the following tabs:

- "User DSN" (if you want to make this data source available only to the user currently logged on to this computer. This data source can only be used on the current machine).

OR

- "System DSN" (if you want to make this data source available to all users from this computer, including NT services).

- 2 Click the **Add** button and choose the 4D v1x ODBC Driver from the list of configured drivers:



- 3 Click **Finish**.

The 4D v1x ODBC Driver's own configuration panel appears:

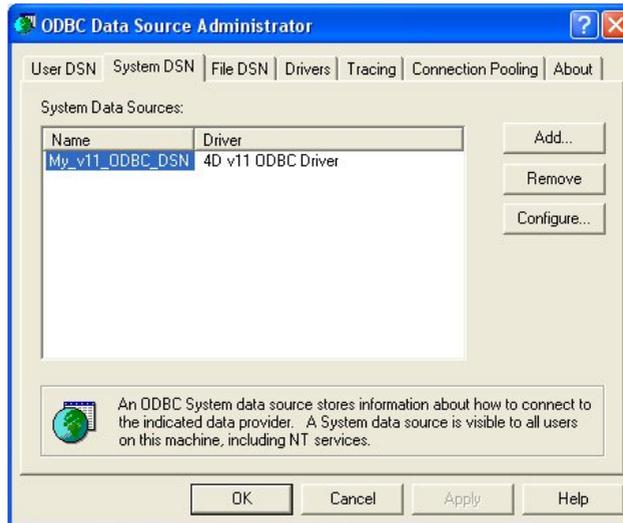


- 4 Click **OK** to save any changes before you close the configuration panel window.

Note : In order to have a full and clear installation, you must not move or rename the installed files after running the 4D ODBC Driver Installer.

Uninstalling on PC Platforms

If you intend to completely remove a 4D v1x ODBC Driver, you must first remove all the Data Source Names created for this driver. This can be done from the ODBC Data Source Administrator:



If you only intend to upgrade a 4D ODBC Driver, you do not need to remove the attached DSNs.

To remove a 4D v1x ODBC Driver, simply launch the 4D v1x ODBC Installer and choose the Remove option; the installer will take care of the rest:



Configuring the Data Source

The fields of these dialog boxes need to be filled in with the following information:

- **Data Source Name:**

Enter a unique name for this Data Source.

Default: "".

This field is mandatory.

- **Description:**

Enter a short description of this Data Source.

Default: "".

This field is optional.

- **Server:**

Enter the 4D network location as: <IP or Name>.

Examples:

"localhost" The 4D application is located on the local machine.

"PC-1" The 4D application is located on the PC-1 machine, published on port 1919 (default port).

"148.1.2.3" The 4D application has the IP address 148.1.2.3.

Default: "localhost".

This field is mandatory.

- **Port:**

Enter the 4D access port.

Default: "1919".

This field is optional.

- **SSL checkbox:**

Checking this option enables connection to the 4D SQL server if the "Enable SSL" option has been set on the SQL/Configuration page of the Preferences.

- **User:**

Enter the user name used to test the connection with 4D.

Default: "".

This field is optional.

■ **Password:**

Enter the password used (in conjunction with the user name) to test the connection with 4D.

Default: "".

This field is optional.

■ **Connection:**

Enter the timeout in seconds to be used (0 means no timeout) after you log in for the driver to connect to a socket on the server:port address.

Default: "".

This field is optional.

■ **Login:**

Enter the timeout in seconds to be used (0 means no timeout) when the driver sends a request and receives the status from the server.

Default: "".

This field is optional.

■ **Query:**

Enter the timeout in seconds to be used (0 means no timeout) when waiting for a response after a query to the server.

Default: "".

This field is optional.

■ **Connection Test button**

Tests the connection with the current parameters.

Uses “User Name” and “Password” fields to login.

Using SSL

Certificate and Key

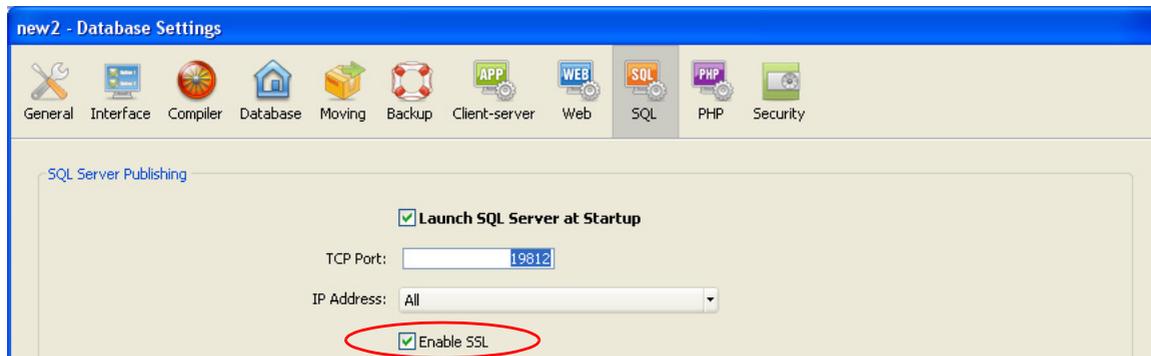
Generating certificates follows the same procedure as for the 4D Web Server (see the “[Using SSL Protocol](#)” command of the *4D Language Reference* manual).

You will obtain two keys: cert.pem and Key.pem. These two files must be put into the Resources/SQL folder of the database.

The certificate files and key only concern the server. It is imperative not to copy them onto any client machines.

Server

On the server side, the use of SSL is set in the Database Settings (SQL page) :



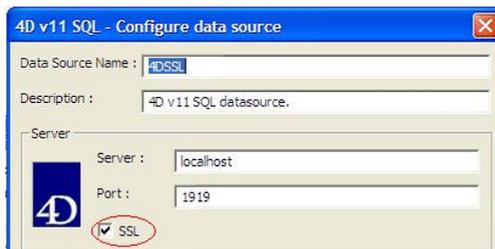
This setting only takes effect on startup of the database. Once you have modified its state (SSL enabled or not) and validated the changes, you will then need to restart your database in order for these changes to be taken into account.

ODBC Client

There are two possibilities depending on whether you use a DSN or a direct connection via SQLDriverConnect.

Connection via a DSN

When configuring the DSN, simply check the “SSL” option in the configuration dialog box:



Connection without a DSN

This occurs when you click on the “Connection test” button which leads to a direct connection via SQLDriverConnect using a connection string.

Here is an example of a connection string that uses SSL:

```
DRIVER="4D v13 ODBC Driver";SSL=true;SERVER=localhost;PORT=1919;UID=Admin
```

This string provides the same information as provided when configuring a DSN. It consists of key/value pairs.

For more information about SQLDriverConnect, please refer to the following address:

<http://msdn2.microsoft.com/en-us/library/ms715433.aspx>

Here is a simple list of keys that can be used in making up the connection string:

- **DRIVER**

Identifies the ODBC Driver used.

Uses the value “4D v1x ODBC Driver”

- **SERVER**

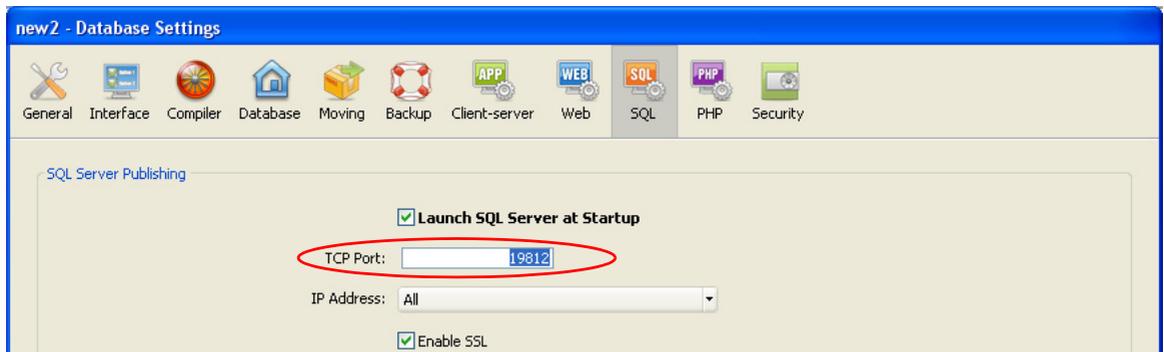
Indicates the IP address or the name of the 4D SQL server to which you want to connect.

Default value: “localhost”.

- **PORT**

Can be used to specify the connection port.

This port must be the same as the one provided on the SQL/Configuration page of the Preferences (on the server machine).



- **UID**

Identifies the user name under which you will be connected.

- **PWD**

Provides the password associated with the user name specified in the UID key.

- **SSL**

Can either be True or False (False by default). Please refer to the section “Using SSL” on page 12.

- **PhysicalConnectionTimeout***

An ODBC connection begins by connecting to the TCP/IP network. This value sets the timeout for this step.

- **LoginTimeout***

Once the driver is connected to the TCP/IP network, it must then be recognized by the 4D SQL server. It thus sends a frame containing the user, password and other information. This timeout sets the maximum time to wait for a reply.

- **QueryTimeout***

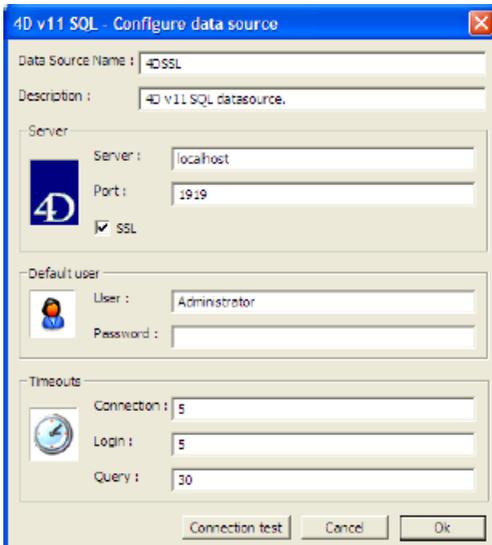
Once an ODBC Driver is connected, it can also execute queries and handle data. This timeout will be applied in this case.

** Timeouts set the length of time to wait before aborting an action. Their values are expressed in seconds, with 0 being no timeout (unlimited wait time).*

- **DSN**

Indicates a DSN to be used. In this case, it is not necessary to use the other keys since they will already be defined in the DSN.

For example, if you have set up a DSN named “4DSSL” as follows:

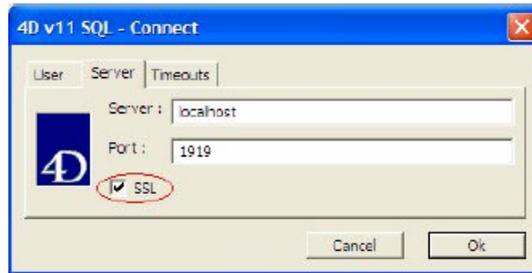


and you then use the following connection string:

```
SQLDriverConnect(ConnectionHandle,WindowHandle,"DSN=4DSSL;
UID=scott;PWD=tiger;SSL=false";SQL_NTS,NULL,0,NULL,
SQL_DRIVER_NOPROMP);
```

Even though your DSN has been configured to use an SSL connection, this connection string will attempt a non-secure connection. In this case, you will be connected as the user “scott” with the password “tiger.”

On the other hand, if you use the SQL_DRIVER_PROMPT option, the following dialog box will appear:



This setting takes effect immediately.

Crystal Reports® Compatibility

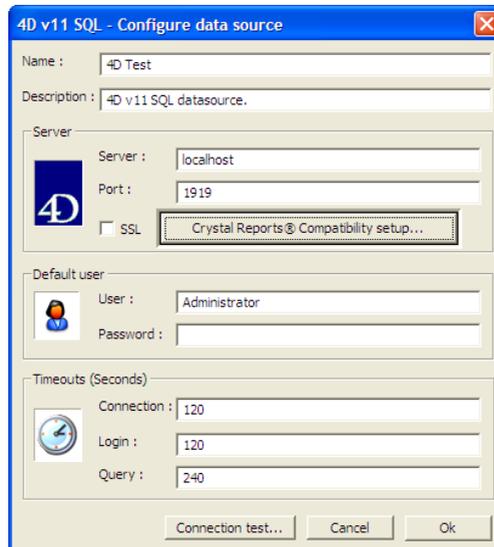
When using 4D with Crystal Reports®, you may encounter problems when building reports from several different tables at once. This is due to the fact that Crystal Reports® does not know how to handle inner and outer joins with 4D.

In order to avoid this problem, there is a simple procedure you can follow:

- 1 Launch the ODBC Driver manager and configure or add a DSN**

This should be done even if you do not need a DSN.

The following dialog box will appear:



2 Press the “Crystal Reports® Compatibility setup...” button

This button only appears for Crystal Report® users. After the setup has been run the first time, it will no longer appear.

After pressing this button, the following dialog box will appear:



3 Press OK to launch the setup procedure.

As mentioned in the dialog box, you may receive warnings from your anti-virus during this procedure. These warnings should be ignored.

4 Close the ODBC Driver configuration window

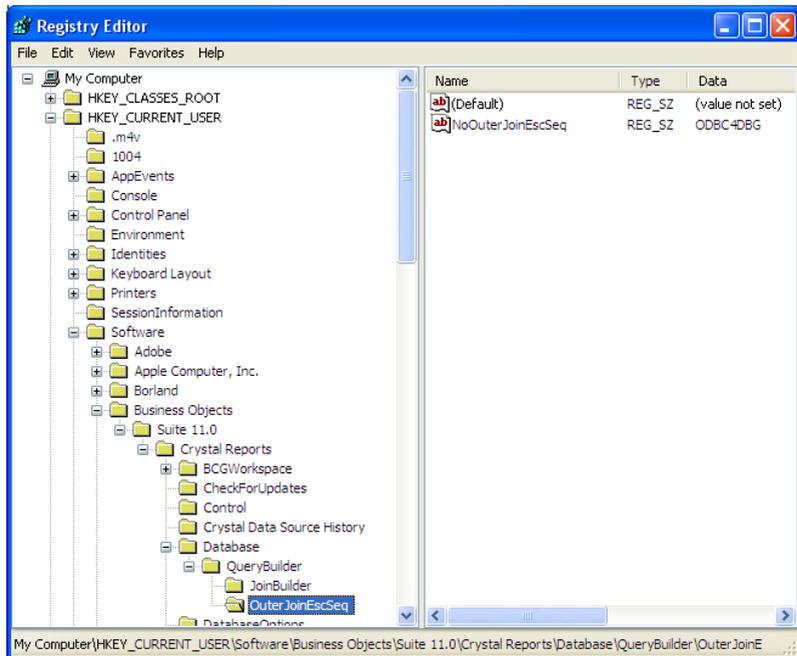
You can press OK, Cancel or simply click on the close box.

5 Restart your machine

You will now be able to use Crystal Reports® without any problems.

How it works

As mentioned in the Compatibility setup dialog box, this procedure writes keys in the registry. If you launch the registry editor utility, you will see something similar to this:



The keys will be updated for Crystal Reports® v10 and v11 as well as their subversions. If you have more than one version installed on your PC, all will be updated.

This is not an overwrite. For example, if you have:

NoOuterJoinEscSeq=MYODBC3

it will become:

NoOuterJoinEscSeq=MYODBC3, ODBC4D

The keys will remain updated even if the 4D ODBC Driver is removed from your machine.

If you install another version or subversion of Crystal Reports®, you will need to go through the compatibility setup procedure again. However, only the newly-installed version will be updated.